SegGAN: Semantic Segmentation with Generative Adversarial Network

1st Xinming Zhang
Beijing Technology and Business University
Beijing, China
izhangxm@foxmail.com

2nd Xiaobin Zhu
Beijing Technology and Business University
Beijing, China
brucezhucas@gmail.com

3rd Xiao-Yu Zhang
Institute of Information Engineering, Chinese Academy of Science
Beijing, China
zhangxiaoyu@iie.ac.cn

4th Naiguang Zhang
Information Technology Institute, Academy of Broadcasting Science
Beijing, China
zhangnaiguang@abs.ac.cn

5th Peng Li
College of Information and Control Engineering China University of Petroleum
Beijing, China
pengli@upc.edu.cn

6th Lei Wang
Information Technology Institute, Academy of Broadcasting Science
Beijing, China
wanglei@abs.ac.cn

***Abstract*—Semantic segmentation is a long standing challenging issue in computer vision. In this paper, a novel method named SegGAN is proposed, in which a pre-trained deep semantic segmentation network is fitted into a generative adversarial framework for computing better segmentation masks. The composited networks are jointly fine-tuned end-to-end to get better segmentation masks. In the pre-training of Generative Adversarial Network (GAN), we try to minimize the loss between the generated images from the generator with the ground truth masks as input and the original images. Our motivation is that the learned GAN shows the relationship between the ground truth masks and the original images, thus the predicted masks of the semantic segmentation model should have the same distribution or relationship with the original images. Concretely, GAN is treated as a kind of loss for semantic segmentation to achieve better performance. Numerous experiments conducted on two publicly available datasets demonstrate the effectiveness of the proposed SegGAN.**

***Index Terms*—Semantic segmentations, Generative adversarial network,**

I. Introduction

Semantic segmentation is one of the most challenging tasks in computer vision, which aim at predicting the pixel-level class label of the input images. In recent years, segmentation approaches based on deep learning methods had become the mainstream in related research domains. As a important task, semantic segmentation is widely used in various applications, i.e., autonomous driving, scene understanding, etc.

Convolutional neural networks (CNNs) have pushed the performance of computer vision systems to soaring heights on a broad array of high-level problems, and have also been adopted in semantic segmentation. However, CNNs are originally designed for the image classification task. To achieve the invariance to numerous transformations, down-sampling operations are frequently conducted. Inevitably, rigid architecture problems are faced. Specifically, the resolution reduction problem restricts the localization accuracy, and the low-resolution feature maps are difficult to predict the boundaries of objects at multiscales. As the pioneer of full CNN-based semantic segmentation work, Fully Convolutional Network (FCN) [15] proposed the skip architecture to solve the resolution reduction problem. In [15], feature maps at lower layers were adopted and upsampled by consecutive deconvolutional operations to produce dense per-pixel labeled outputs. In [12], a visual-attention-aware model is proposed to mimic the human visual system for salient-object detection. In [13], the algorithm, Quaternionic Distance Based Weber Descriptor (QDWD) which was initially designed for detecting outliers in color images, is used to represent the directional cues in an underwater image. In [17] [2], an encoder-decoder architecture was proposed to recover the spatial information from the low-resolution feature maps. In deeplab [4], atrous convolutional operations were adopted to solve the feature resolution reduction problem. The downsampling operators were removed from the last few max pooling layers, and atrous convolutions were adopted in the subsequent convolutional layers, to generate lager feature maps. In [3] [18], the atrous spatial pyramid pooling (ASPP) operation was proposed to solve the multiscale problem during segmentation. In [14] [20], DenseCRF was adopted to capture long range information to refine the predicted boundaries. Although, numerous solutions were proposed in semantic segmentation related domain , the problems of accurate boundary extraction and areas prediction still remain to be the challenging tasks.

In this paper, we concentrate on modeling the distribution of the statistical relationship between the predicted masks and input images. The motivation is that a good segmentation mask should have strong correlation with the input image. With the help of Generative Adversarial Network (GAN) [8], we can even generate a similar image from a good mask. Thus, we attempt to learn a GAN, in which the inputs are the ground truth masks. And we try to minimize the loss between the generated images and the real images. Then the learned GAN is treated as a kind of loss for semantic segmentation

978-1-5386-5321-0/18/$31.00

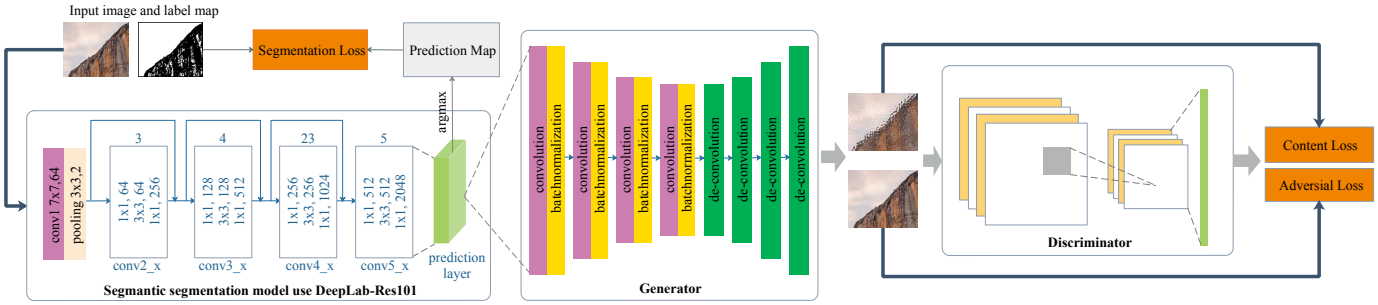

Fig. 1. The proposed framework for SegGAN.

model for finetuning. The proposed method is named as SegGAN, and is composed of three key components: (1) a semantic segmentation model, (2) a generator, for generating images from the predicted masks, (3) a discriminator, for distinguishing original images from dataset or fake images reconstructed by generator. Note that, the GAN model need to be trained firstly. We adopt the ground-truth images as the input of the generator and try to minimize the loss between the generated images and the original images. The goal of this step is to learn the relationship between the ground truth masks and the original images. Then, the GAN model is combined with the segmentation model. In detail, the outputs of the semantic segmentation network are directly put into pre-trained generative model. GAN can learn the distribution p_z of the ground-truth label data in an adversarial way.

Our main contributions can be summarized as follows:

- A novel method named SegGAN is proposed, in which the learned GAN is adopted to refine the segmentation masks.
- We adopt GAN to learn the relationship between the masks and the real images;
- The proposed method achieves promising performance against the state-of-the-art works.

In Section II, we present our approach, where we first describe the design and structure, then cover the training steps. Section III deals with experimental results, where we report our results on Pascal VOC 2012 [6], and StanfordBG datasets [9]. Finally, we conclude the paper in Section IV.

II. Proposed Method

A. Architecture

The proposed SegGAN (as shown in Fig. 1) is a differentiable method in which the GAN model is combined with an existed segmentation network. The proposed hybrid framework is optimized in an end-to-end manner. The SegGAN consists of semantic segmentation model, discriminator and the generator. The task of our generator network G is to generate an image based on the predictions layer of the segmentation model.

Following the architecture in [11], we define generator network G and discriminator network D in such a way that D can act as supervisor to G in the min-max optimization process. This process aims at training the generator G to be able to generate a synthetical image I^R corresponding with the original one, while the discriminator D tries to distinguish between the original image and the reconstructed image I^R. The detail loss function of GAN can be defined as follows:

$$\min_{\theta_{\mathbf{G}}} \max_{\theta_{\mathbf{D}}} \log\left(\mathbf{D}\left(\mathbf{I}\right)\right) + \log\left(1 - \mathbf{D}\left(\mathbf{I}^R\right)\right) \tag{1}$$

The generator takes simultaneously output of predictions layer p_{seg} and original image I as input, then try to generate same image with original image I. This processing can be described as $G(p_{seg})$. Highly motivated by [11] [19] [22], the G network in the proposed method adopts 4 convolutional layers and 4 de-convolutional layers with random dropout at rate the 0.5 to avoid overfit trap. In addition, the D network adopts 4 convolutional layers with ReLU method as activation function followed every layer but excludes the last layer. The original image and the fake image are sent into the D network concurrently. So that the random process of selecting an image from them could be avoided.

In the proposed method, DeepLab [4] is selected as a basic segmentation model for the below two reasons: (1) Computation performance: the fully-connected CRF generally requires 0.5 second per image while DeepLab only need 0.12 second. (2) Simplicity:A serious of methods, such as CRF and CNNs, can well combined with DeepLab as post-precessing functions. The goal of segmentation model is to generat confidence maps $p_{seg} \in R^{c*w*h}$, where the c is dataset class number and w, h are the width and height of prediction maps respectively. Then, the argmax operation will be performed to the predictions layer to get the final prediction mask, which each value show the label of response pix of the input image. In detail, the segmentation model has 4 blocks and 4 dilation layers without fully connect layer to accomodate different input size. Without loss and generality, we use fix size input image by pre-precessing, such as random crop or resize.

B. Optimization

The loss function ℓ in our method consists of three main key componments: segmentation loss ℓ_S, content loss ℓ_C and adversarial loss ℓ_A. In detail, the loss function ℓ can be formulated as follow:

$$\ell = \ell_S + \lambda_1 \ell_C + \lambda_2 \ell_A \tag{2}$$

TABLE I
PER-CLASS RESULTS ON THE PASCAL VOC 2012 VALIDATION SET. THE REFERENCE OF THE METHOD MARKED THE '†' CAN BE FOUND AT [21]

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mIoU
FCN [15]	76.8	34.2	68.9	49.4	60.3	75.3	74.7	77.6	21.4	62.5	46.8	71.8	63.9	76.5	73.9	45.2	72.4	37.4	70.9	55.1	62.2
Zoom-out †	85.6	37.3	83.2	62.5	66.0	85.1	80.7	84.9	27.2	73.2	57.5	78.1	79.2	81.1	77.1	53.6	74.0	49.2	71.7	63.3	69.6
CRF-RNN †	87.5	39.0	79.7	64.2	68.3	87.6	80.8	84.4	30.4	78.2	60.4	80.5	77.8	83.1	80.6	59.5	82.8	47.8	78.3	67.1	72.0
DeconvNet [17]	89.9	39.3	79.7	63.9	68.2	87.4	81.2	86.1	28.5	77.0	62.0	79.0	80.3	83.6	80.2	58.8	83.4	54.3	80.7	65.0	72.5
GCRF [14]	† 85.2	43.9	83.3	65.2	68.3	89.0	82.7	85.3	31.1	79.5	63.3	80.5	79.3	85.5	81.0	60.5	85.5	52.0	77.3	65.1	73.2
DPN †	87.7	59.4	78.4	64.9	70.3	89.3	83.5	86.1	31.7	79.9	62.6	81.9	80.0	83.5	82.3	60.5	83.2	53.4	77.9	65.0	74.1
Piecewise †	90.6	37.6	80.0	67.8	74.4	92.0	85.2	86.2	39.1	81.2	58.9	83.8	83.9	84.3	84.8	62.1	83.2	58.2	80.8	72.3	75.3
PSPNet [21]	91.8	**71.9**	**94.7**	71.2	75.8	95.2	89.9	**95.9**	39.3	90.7	71.7	90.5	**94.5**	88.8	89.6	72.8	89.6	64.0	85.1	76.3	82.6
DeepLab [4]	90.1	59.3	90.4	85.1	89.4	94.9	91.4	94.0	65.6	92.1	84.8	92.8	89.4	87.4	88.9	77.5	91.0	**84.5**	92.1	**87.3**	86.9
SegGAN	**92.1**	46.5	**92.2**	**86.5**	**90.4**	**96.0**	**92.5**	**95.6**	**65.9**	**94.1**	**82.7**	**94.4**	**92.1**	**89.5**	**90.5**	**79.1**	**93.2**	83.2	**94.1**	87.2	**87.4**

where the λ_1 and λ_2 are two emprical weight parameters.

In our method, the multi-label cross-entropy loss [10] is adopted to evaluate the performance of segmentation performance. The nature of segmentation is a dense classification task. Consequently, the cross-entropy loss, instead of classification loss and mean squared loss, is more suitable for neural network classifier training. The detail segmentation loss function is defined as:

$$\ell_S = \sum_{x}^{N} \sum_{i}^{C} P_{xi} \log(Y_{xi}) \tag{3}$$

where the P_{xi} is computed by segmentation model which indicates the probability of assigning label i to pixel x, and Y_{xi} indicates the probability of ground-truth label.

The content loss is used to calculate quality of the synthetical images I^R reconstructed by G network. The pixel-wise MSE loss is the widely used criterion, and calculated as:

$$\ell_C = \frac{1}{WH} \sum_{i=1}^{W} \sum_{j=1}^{H} \left(I_{ij} - I_{ij}^R\right)^2 \tag{4}$$

The adversarial loss reflects the quality of synthetical image I^R reconstructed by G, which is defined as:

$$\ell_A = \log\left(\mathbf{D}\left(\mathbf{I}\right)\right) + \log\left(1 - \mathbf{D}\left(\mathbf{I}^R\right)\right) \tag{5}$$

III. EXPERIMENTS

A. Experimental settings

All the networks are implemented based on the TensorFlow framework [1]. The SegGAN framework is trained in two steps: Firstly, the adversarial generative network is trained to learn the distribution of the ground truth masks. Limited to the GPU memory, the batch size is set as 8. As for the training of the generator, the Adam optimizer is adopted with isotropic Gaussian weights. The AdamOptimizer's learning_rate, beta1, beta2 and epsilon are set as 0.001, 0.9, 0.999, 1e-08 respectively. The loss function applied to this optimization process can be formulated as:

$$\ell_G = 100 * \ell_C + \ell_A \tag{6}$$

After 20000 iterations learning, we finish the training of our GAN model.

In terms of the training on PASCAL VOC 2012, the pretrained parameters provided by DrSleep [5] are directly adopted. For the training of the segmentation model on the StanfordBG Dataset, the parameters are initialized using the weights pretrained on ImageNet.

After the GAN model and the segmentation model have been fine-tuned, we combine the segmentation model, generator model and discriminator model as a whole framework. Note that, the weights in the layers of the segmentation network are initialized in the same way as the previous step. The standard Adam optimizer is utilized for the optimization of the semantic segmentation model, and the adversarial networks are initialized using pre-trained weights from the first step.

As for the loss function formulated in formula (2), the λ_1 and λ_2 are both set as 0.1. Then, the optimization of the segmentation model is conducted based on the adopted loss function. Note that, the parameters of the GAN are not update. Similar as the training process of the generator, the SegGAN is also trained for 20000 iterations. In the testing phase, we only adopted the semantic segmentation layers to conduct mask prediction.

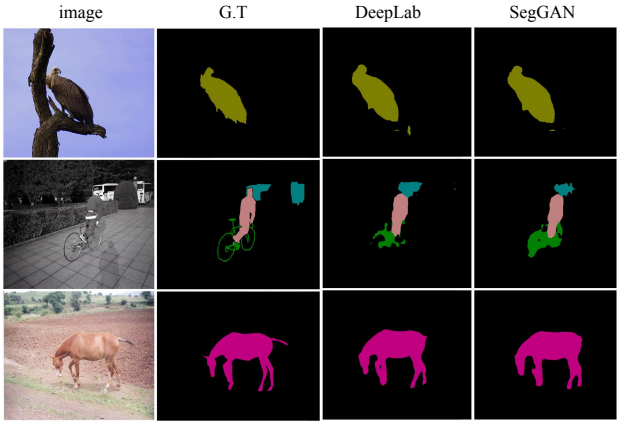

Fig. 2. Samples of images produced by segmentation model on Pascal VOC.

B. Experiments on the PASCAL VOC 2012

In PASCAL VOC 2012 [6], there are 20 foreground object classes and one background class. The original dataset contains 1464, 1449, and 1456 images for training, validation, and

TABLE II
THE SEGMENTATION PERFORMANCE ON THE STANFORDBG DATASET. THE REFERENCE OF THE METHOD MARKED THE '†' CAN BE FOUND AT [7].

Method	Class Acc	Pix Acc	Mean IoU
Gould et al. 2009†	-	76.4	-
Munoz et al. 2010†	66.2	76.9	-
Tighe et al. 2010†	-	77.5	-
Socher et al. 2011†	-	78.1	-
Kumar et al. 2010†	-	79.4	-
Lempitzky et al. 2011†	72.4	81.9	-
multiscale convnet†	72.4	78.8	-
INRIA et al. 2016 [16]	68.7	75.2	54.3
DeepLab [4]	75.9	87.0	63.4
SegGAN	**79.0**	**89.3**	**69.9**

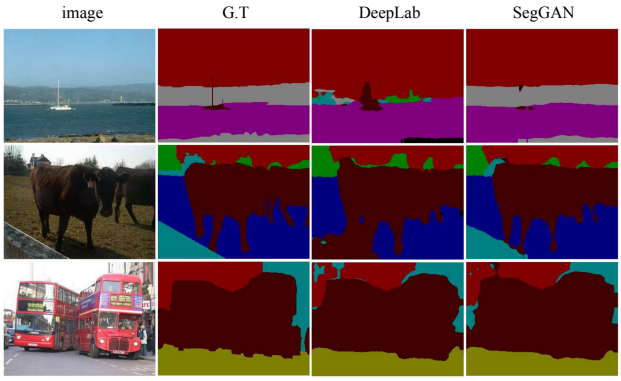

Fig. 3. Visual comparisons on the StanfordBG Database.

testing, respectively. The dataset is augmented by the extra annotations, resulting in 10582 training images.

We evaluate the performance of the standard segmentation model and the SegGAN framework on the PASCAL validation set, and our model is trained on the augmented PASCAL training set. The identification results are presented in Table I. The performance of different models are measured by the mean Intersection over Union (IoU) proposed in [15], and it is calculated as:

$$S_{mIoU} = (1/n_{\text{cl}}) \sum_i n_{ii} / \left(\sum_j n_{ij} + \sum_j n_{ji} - n_{ii} \right) \quad (7)$$

where the n_{ii} is the number of pixels of class i predicted to class i, n_{cl} is the number of the dataset classes.

By adopting GAN in DeepLab, the proposed SegGAN can achieve a higher mIoU score than the original DeepLab. It indicates that more present classes in the images are identified correctly. Admittedly, the proposed model doesn't get best performance at some classes, such as bike, sofa and tv. The reason why our model get worse performance on those classes is that the GAN model's fully-connected layer eliminate the location information from the original image. Despite having those defects, SegGAN achieves the best result under this train/test protocol for Pascal VOC dataset. The visual comparisons are provided in Fig.2. The original images, the ground truth, the results of DeepLab and the results of SegGAN are shown in Fig. 2 from the first column to the fourth column. These images clearly indicate that our GAN model is able to learn hidden structures, and can be adopted to enhance the performance of our segmentation model.

C. Experiments on the Stanford Background dataset

Experiments are also conducted on the Stanford Background dataset(StanfordBG) introduced in the work of [9]. The dataset contains 715 images, and we random select 600 for training and 115 images for testing. The experimental results are shown in Table II. The symbol '-' means the paper has no data of this term. Since some methods have no mIoU data, we use the pixel accuracy and class accuracy defined in [15] to measure the performance.

SegGAN can also achieve the best performance on the StandforBG dataset. It can be seen that the class accuracy is improved from 75.9% to 79.0%. And the mIoU accuracy is improved from 63.4% to 69.9%. Sampled predicted masks provided by different segmentation models are shown in Fig.3. Similar to Fig.2, these images clearly suggest that GAN model can effectively detect the perturbing of input data and feed back it to the segmentation model.

IV. CONCLUSIONS

In this paper, we propose a novel framework called SegGAN for the semantic segmentation task. In SegGAN, the GAN is adopted to dig out the relationship between the masks and images, and then the learned GAN is treated as a kind of loss to finetune the semantic segmentation model. Numerous experiments on two publicly available datasets show that the SegGAN can well improve the performance of the semantic segmentation model.

V. ACKNOWLEDGMENTS

The corresponding authors of this work are Xiaobin Zhu (Email: brucezhucas@gmail.com) and Xiao-Yu Zhang (Email: zhangxiaoyu@iie.ac.cn). This work was supported by National Key R&D Program of China (2017YFB1401000) and National Natural Science Foundation of China (61501457, 61602517).

REFERENCES

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning." in *OSDI*, vol. 16, 2016, pp. 265–283.

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *arXiv preprint arXiv:1511.00561*, 2015.

[3] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs," *arXiv.org*, Dec. 2014.

[4] ——, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2018.

[5] DrSleep, "Deeplab-resnet rebuilt in tensorflow," 2017, https://github.com/DrSleep/tensorflow-deeplab-resnet/.

[6] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *International journal of computer vision*, vol. 111, no. 1, pp. 98–136, 2015.

[7] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1915–1929, 2013.

[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[9] S. Gould, R. Fulton, and D. Koller, "Decomposing a scene into geometric and semantically consistent regions," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 1–8.

[10] H. He and R. Xia, "Joint binary neural network for multi-label learning with applications to emotion classification," *arXiv preprint arXiv:1802.00891*, 2018.

[11] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *arXiv preprint*, 2017.

[12] M. Jian, K. Lam, J. Dong, and L. Shen, "Visual-patch-attention-aware saliency detection," *IEEE Trans. Cybernetics*, vol. 45, no. 8, pp. 1575–1586, 2015. [Online]. Available: https://doi.org/10.1109/TCYB.2014.2356200

[13] M. Jian, Q. Qi, J. Dong, Y. Yin, and K. Lam, "Integrating QDWD with pattern distinctness and local contrast for underwater saliency detection," *J. Visual Communication and Image Representation*, vol. 53, pp. 31–41, 2018. [Online]. Available: https://doi.org/10.1016/j.jvcir.2018.03.008

[14] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *Advances in neural information processing systems*, 2011, pp. 109–117.

[15] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[16] P. Luc, C. Couprie, S. Chintala, and J. Verbeek, "Semantic Segmentation using Adversarial Networks." *CoRR*, 2016.

[17] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1520–1528.

[18] J. Yao, S. Fidler, and R. Urtasun, "Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 702–709.

[19] C. Zhang, Z. Xue, X. Zhu, H. Wang, Q. Huang, and Q. Tian, "Boosted random contextual semantic space based representation for visual recognition," *Inf. Sci.*, vol. 369, pp. 160–170, 2016. [Online]. Available: https://doi.org/10.1016/j.ins.2016.06.029

[20] X. Zhang, S. Wang, and X.-c. Yun, "Bidirectional active learning: A two-way exploration into unlabeled and labeled data set." *IEEE Trans. Neural Netw. Learning Syst.*, vol. 26, no. 12, pp. 3034–3044, 2015.

[21] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2881–2890.

[22] X. Zhu, J. Liu, J. Wang, C. Li, and H. Lu, "Sparse representation for robust abnormality detection in crowded scenes," *Pattern Recognition*, vol. 47, no. 5, pp. 1791–1799, 2014. [Online]. Available: https://doi.org/10.1016/j.patcog.2013.11.018